

## **REMARKS**

Claims 1-34 are pending in the application. Claims 1, 7, 17, 21 and 31 are independent claims. Claims 1-34 stand rejected. Further review and consideration is respectfully requested in view of the amendments and following remarks.

### ***Claim Rejections – 35 USC § 112***

Claims 1 – 34 stand rejected under 35 U.S.C. 112, second paragraph for allegedly being indefinite. Without conceding the propriety of the rejection, Applicants have amended claims 1, 7, 17, 21, and 31. Accordingly, Applicants respectfully submit that the rejection is overcome.

### ***Claim Rejections – 35 USC § 102/103***

Claims 1-34 stand rejected as either anticipated over U.S. Patent Application No. 10/268,856 to Khodabakhian or obvious over Khodabakhian in view of U.S. Patent Application No. 10/026,932 to Hogg or Khodabakhian in view of Hogg and further in view of U.S. Patent Application No. 09/815286 to Jones. Applicants respectfully traverse these rejections.

#### **Claim 1**

Claim 1 stands rejected under 35 U.S.C. § 102(e). In the Office Action the Examiner stated that the “Exception handler 202/Debugger 204 [of] [] paragraphs 0025-0027” (Office Action at p. 3) teaches the limitation “wherein the response processing code has failure handling functionality.” Applicants disagree. According to claim 1, the instance of automated business process includes response processing code, the response processing code includes exception handling code. Applicants respectfully submit that this aspect is missing from Khodabakhian. According to the portions of Khodabakhian cited by the Examiner the orchestration server 102 includes an exception handler 202 that processes exceptions generated by orchestration engine 200 when implementing the instructions and commands in a scenario 110. (See paragraph [0025] and FIG. 2). Applicants submit that the exception handler 202 and associated debugger 204 are not included in a scenario 110 and this

configuration fails to teach or suggest response processing code of an automated business process instance that includes exception handling code as recited in claim 1. Applicants respectfully submit that the configuration of Khodabackchian, i.e., having error handling code outside of an instance, adds complexity to the automation of business processes and is a shortcoming cured by the instant application since a developer using techniques described in Khodabackchian would have to “keep track of a variety of program states to enable the compensation handler 320 can carry out the proper steps while operating independently from the main process 310. In addition, a developer must incorporate a method for the compensation handler 320 and main process 310 to interact after the compensation actions have been completed so the main process 310 will know the proper state(s) of the program.” (Application at paragraph [0057]). Accordingly, for at least this reason Applicants respectfully request reconsideration of the rejection of claim 1.

Insomuch as claims 2, 3, 5, and 6 depend directly from claim 1 they too define over the cited art for similar reasons described above with respect to claim 1. Accordingly, Applicants respectfully request reconsideration of the rejections of claims 2, 3, 5, and 6.

#### Claim 7

Claim 7 stands rejected under 35 U.S.C. § 102(e) over Khodabakchian. In the Office Action the Examiner stated that the limitation “processing the response using response processing code within the instance according to the success or failure of the message, wherein the response processing code has failure handling functionality” (Claim 7 emphasis added) is taught by “(Exception Handler 202/Debugger 204 page paragraphs 0025-0027)” (Office Action at page 5). Applicants traverse this rejection. According to claim 7, the response processing code that include failure handling functionality is within the instance. This aspect is not taught by Khodabakchian. According to the portions of Khodabakchian cited by the Examiner the orchestration server 102 includes an exception handler 202 that processes exceptions generated by orchestration engineer 200 when implementing the instructions and commands in a scenario 110. (See paragraph [0025] and FIG. 2). Applicants submit that the exception handler 202 and associated debugger 204 are not included in a scenario 110. Applicants respectfully submit that the configuration of Khodabackchian, i.e., having error handling code outside of an instance, adds complexity to

the automation of business processes and is a shortcoming cured by the instant application. Accordingly, for at least this reason Applicants respectfully request reconsideration of the rejection of claim 7.

Insomuch as claims 8 – 16 depend directly or indirectly from claim 7 they too patentably define over the cited art for at least similar reasons as claim 7. Accordingly, Applicants respectfully request reconsideration of the rejections of claims 8-16.

Independent claims 17, 21, and 31 include similar limitations to those in claim 7 and patentably define over the art of record for at least the reason described above with respect to claim 7. Accordingly, Applicants respectfully request reconsideration of the rejection of claims 17, 21, and 31.

Insomuch as claims 18 – 20, 22-30, and 32-34 depend directly or indirectly from claims 17, 21, or 31 they too patentably define over the cited art for at least similar reasons as claim 7. Accordingly, Applicants respectfully request reconsideration of the rejections of claims 18 – 20, 22-30, and 32-34.

***Rejection of Claims under 35 U.S.C. § 103(a) in view of Khodabakhian & Hogg***

The Office Action rejected claims 2, 8-10, 17-20, 22-24, 32, and 33 under 35 U.S.C. § 103(a) as allegedly obviated by Khodabakhian in view of Hogg. In addition to the reasons described above Applicants traverse the rejections of claims 2, 8-10, 17-20, 22-24, 32, and 33 for at least these additional reasons.

As previously noted, Khodabakhian fails to disclose “processing the response using response processing code within the instance according to the success or failure of the message, wherein the response processing code has failure handling functionality.” Also, the Office Action admits that Khodabakhian fails to disclose a try-catch block in the processing code. (Office Action at p. 6).

Hogg discusses a synchronous communication, *i.e.*, so-called RPC, system, Hogg at ¶¶ 0096-0099, to “maintain a central database from a number of distributed locations,” Hogg at ¶0025, in which Hogg states that “[k]eeping transactions short and synchronous permits the entire database to be locked during each transaction.” Hogg, Abstract. “As shown in FIG. 2,

[the] system . . . has three major software-based components, client components 22 which run on user computers 12, a server component 30 and a remote procedure call (RPC) system 26 for maintaining communications between client components 22 and server component 30.” Hogg at ¶0028. “RPC system 26 should report errors.” Hogg at ¶0138. “RPC system 26 can preferably report these errors by throwing an exception.” Hogg at ¶0143. “Preferably software client component 22 includes a canonical event routine handling pattern which includes a try-catch block to deal with these expected errors when they occur.” Hogg at ¶0144. “The catch part of the try-catch block includes statements which deal with the errors. These statements may use the exception type to distinguish between errors of different types.” Hogg at ¶0148.

The Office Action alleges that it would have been obvious to one of ordinary skill in the art to: (a) select Khodabakhchian for its asynchronous system with an independent centralized exception handler, (b) choose to modify failure handling in Khodabakhchian despite already having an independent centralized exception handler, (c) select Hogg out of the multitude of references for its synchronous system with client component try-catch block, (d) and choose to supplement, or replace Khodabakhchian’s independent centralized exception handler with a try-catch block in each process running on Khodabakhchian’s Web Service Orchestration Server. (Office Action at p. 7). The Office Action alleges b-d were obvious because a try-catch block in each process would allow Khodabakhchian to distinguish between errors of different types. *Id.*

It is respectfully submitted that the reasoning of the Office Action to combine and modify is without merit at least because Khodabakhchian’s independent centralized exception handler already allowed Khodabakhchian to distinguish between errors of different types. The Examiner has failed to provide a reason as to why one of skill would modify a system that already has a centralized exception handler to include a try-catch exception handler. The reasoning provided by the Examiner, i.e., “a try-catch block in each process would allow Khodabakhchian to distinguish between errors of different types” is already solved by the exception handler of Khodabakhchian. (See, e.g., paragraph [0025] stating “[e]xceptions may include, for example, a fault generated by a web service or a notice generated as a result of a

**DOCKET NO.:** MSFT-2748/302029.1  
**Application No.:** 10/698,762  
**Office Action Dated:** January 2, 2008

**PATENT  
REPLY FILED UNDER EXPEDITED  
PROCEDURE PURSUANT TO  
37 CFR § 1.116**

web service timeout.”) For this additional reason, Applicants respectfully request reconsideration of the rejections of claims 2-6, 8-16, 18-20, 22-30 and 32-34.

### **CONCLUSION**

Applicants request the Examiner reconsider the rejections and issue a Notice of Allowance of all the claims.

Date: April 2, 2008

/David M. Platz/  
David M. Platz  
Registration No. 60,013

Woodcock Washburn LLP  
Cira Centre  
2929 Arch Street, 12th Floor  
Philadelphia, PA 19104-2891  
Telephone: (215) 568-3100  
Facsimile: (215) 568-3439